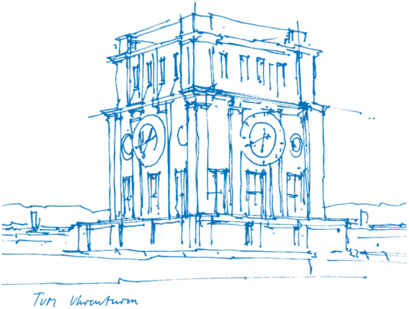
Analysis of Machine Learning for State Register Identification

Lee Seng Hwee
Technische Universität München
Faculty of Electrical and Computer Engineering
Institute for Security in Information Technology

University
June 27, 2020



Outline

Introduction

The need to Identify State Registers:

Due to an increase in the reliance on 3^{rd} parties, especially in the manufacturing field, there is high risk of logics not implemented by the desginers being implemented in the the manufaturing, which can lead to

- Loss of Privacy
- Loss of Data
- System Failures

Therefore there is a need to reverse engineer the manufactured design and identify these logics to mitigate the risk.

Introduction

Current Approach

Traditional approach to Identitfy State Registers compares the extracted netlist from the manufactured product to a golden model

- Limitation: Requires golden model

RELIC and fastRELIC were developed to overcome this limitation of requiring a golden model by assgining similarity scores. Registers with higher similarity scores are less likely to be State Registers.

- Limitation: Requires human input to classify registers

Problem Statement

Both tradtional and RELIC/fastRELIC approaches can be tedious and may be subjected to human influence/error.

Proposed Solution

To train and deploy a machine learning model for State Register identification.

Advantages of Machine Learning:

- Does not require human to interpret
- Faster Processing
- Ease of use
- Consistency in evaluation

The Data

Prior to creating a Neural Network for State Register Identification, 5 methods of implemention of the feature file was discussed for training the Neural Network

- Original feature set
- Original feature set with Cosine Similarity Score
- Original feature set with Euclidean Distance Similarity Score
- Original feature set with fastRELIC Similarity Score
- Original feature set with all the addition features

The Additional Features

Cosine Similarity Score

- Register slices of a certain depth from the designs files were extracted into a vector
- Register slices vectors were compared to each other, producing a similarity score using cosine similarity

$$cos_{sim}(u, v) = cos(\theta) = \frac{u \cdot v}{|u||v|} \tag{1}$$

where u and v are n^{th}-dimensional vectors

The Additional Features

Euclidean Distance Similarity Score

- Register slices of a certain depth from the designs files were extracted into a vector
- Register slices vectors were compared to each other, producing a similarity score using Euclidean Distance

$$E(u, v) = \sqrt{\sum_{i=1}^{n} (u_i - v_i)^2} \quad (2)$$

where u and v are n^{th}-dimensional vectors

The Additional Features

fastRELIC Similarity Score

- Register slices of a certain depth from the designs files were extracted into a vector
- fastRELIC's Pair Similarity scores algorithm was used to assign a similarity score between 0 to 1

Features Selection

Constant Filtering

- Removing features with constant values

Quasi-constant filtering

- Removing features with a value diffence less than a selected threshold

Feature Permutation

- Randomise the feature values
- Train on pre-optimised neural network
- Compare permuted accuracy with unpermuted accuracy

Sequential Feature Selection

- Train on pre-optimised neural network
- Select the best feature combination based on accuracy by adding features one at a time

Addtional Features Testing Methodology

- 12 files for training
- 1 file for testing
- Per file was used to train the model 100 times
- Experiment repeated 5 times
- Average results(Model accuracy, State Register accuracy) per implementation across all test

$$\mathrm{A} = \frac{\text{Number of Correctly Predicted Registers}}{\text{Total Number of Registers}} \tag{3}$$

$$\mathrm{SRA} = \frac{\text{Number of Correctly Predicted State Registers}}{\text{Total Number of State Registers}} \tag{4}$$

Feature Permutaion Methodology

- 12 files for training
- 1 file for testing
- Each feature in a file permuted individually and train the model for 100 times
- Average the 100 accuracies per features
- Train model with unpermuted data set for 100 times and calculate average
- Repeat experiment for 5 times
- Calculate Ratio(Method 1) and Feature Occurence(Method 2)

Ratio — Method 1

$$R_n(A_{original}, A_{permuted}) = \frac{A_{original}}{A_{permuted}} \tag{5}$$

$$SRR_n(SRA_{original}, SRA_{permuted}) = \frac{SRA_{original}}{SRA_{permuted}} \tag{6}$$

$R_n < 1$	$SRR_n < 1$	Feature Hindrance
$R_n = 1$	$SRR_n = 1$	Feature Hindrance
$R_n > 1$	$SRR_n > 1$	Feature Important

Table: Ratio Interpretation

Feature Occurence — Method 2

$$C(n) = \left(\frac{1}{k}\sum_{i=0}^{k}[f_i = n]\right) \tag{7}$$

$A_{original} < A_{permuted}$	Feature Hindrance
$A_{original} = A_{permuted}$	Feature Hindrance
$A_{original} > A_{permuted}$, with a difference of $> 1\%$	Feature Important

Table: Conditions for filtering

Sequential Feature Selection

- Run 5 times per file per implementation
- Count occurence per feature across all files
- Normalize

Results — Implemetation of addition features

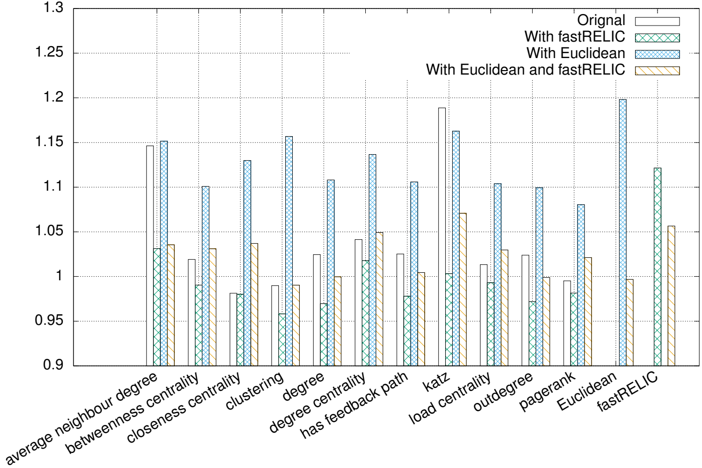
Implementation	Model Accuracy Average
Original	0.75
With fastRELIC	0.77
With Euclidean	0.83
With Euclidean and fastRELIC	0.83

Table: Model Accuracy

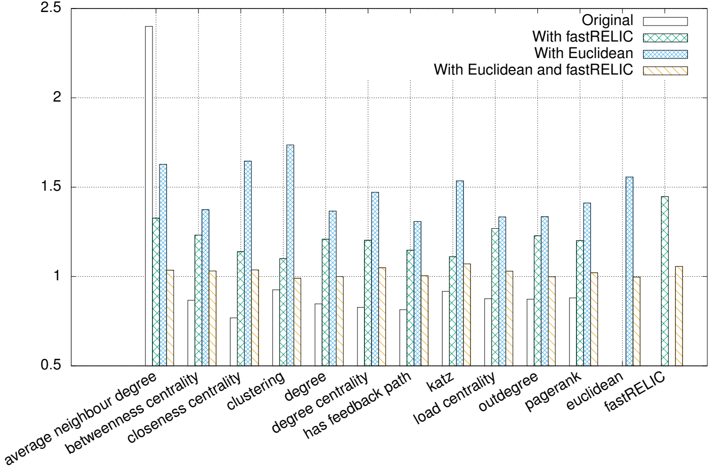
Implementation	State Register Accuracy Average
Original	0.59
With fastRELIC	0.65
With Euclidean	0.71
With Euclidean and fastRELIC	0.74

Table: Model Accuracy: State Register Prediction

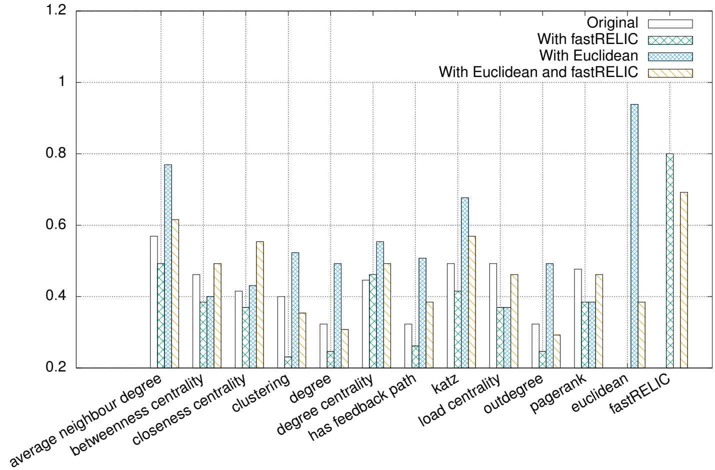
Results – Feature Permutation Method 1
Model Accuracy Ratio



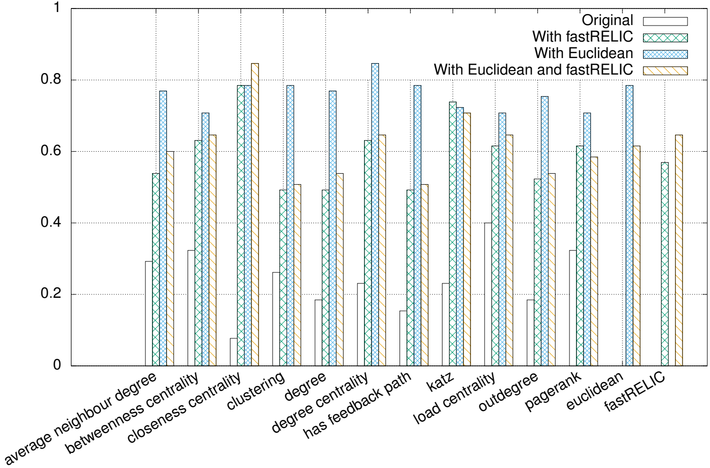
Results – Feature Permutation Method 1
State Register Accuracy Ratio



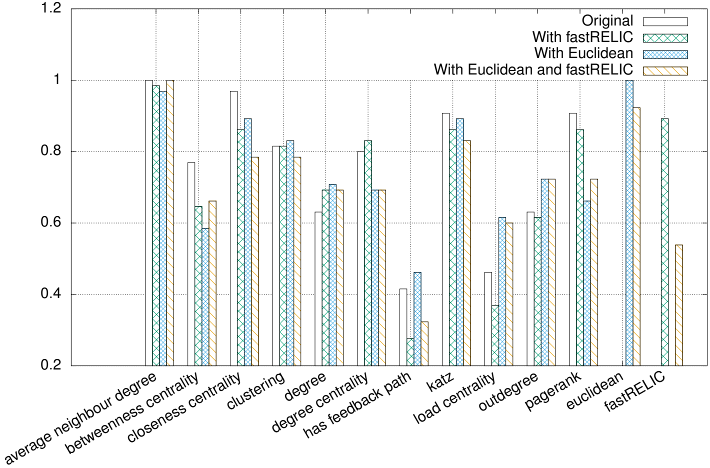
Results – Feature Permutation Method 2
Model Feature Occurance



Results – Feature Permutation Method 2
State Register Feature Occurance



Results – Sequential Feature Selection



Results — Removing Features

Implementation	Model Accuracy Average
Original	0.75
With fastRELIC	0.78
With Euclidean	0.83
With Euclidean and fastRELIC	0.83

Table: Model Accuracy

Implementation	State Register Accuracy Average
Original	0.59
With fastRELIC	0.65
With Euclidean	0.70
With Euclidean and fastRELIC	0.74

Table: Model Accuracy: State Register Prediction